

Exam Logical Verification

May 27, 2011

There are six (6) exercises.

Answers may be given in Dutch or English. Good luck!

Exercise 1. (*5+5+5+4 points*)

This exercise is concerned with first-order propositional logic (**prop1**) and simply typed λ -calculus ($\lambda \rightarrow$).

- Show that the following formula is a tautology of minimal **prop1**:
 $((A \rightarrow B) \rightarrow C) \rightarrow B \rightarrow C$.
- Give the type derivation in $\lambda \rightarrow$ corresponding to the proof of 1a.
- Give closed inhabitants in normal form of the following types
(NB: it is not asked to give the type derivations):

$$\begin{aligned} &(A \rightarrow B) \rightarrow (B \rightarrow C) \rightarrow A \rightarrow C \\ &A \rightarrow B \rightarrow A \\ &((A \rightarrow B \rightarrow A) \rightarrow A) \rightarrow A \end{aligned}$$

- Replace in the following three terms the ?'s by simple types, such that we obtain typable λ -terms (NB: it is not asked to give the type derivations).

$$\begin{aligned} &\lambda x:?. \lambda y:?. y x x \\ &\lambda x:?. \lambda y:?. x (\lambda z:?. y) y \end{aligned}$$

Exercise 2. (*5+3+5 points*)

This exercise is concerned with first-order predicate logic (**pred1**) and λ -calculus with dependent types (λP).

- Show that the following formula is a tautology of minimal **pred1**:
 $(\forall x. P(x) \rightarrow Q(x)) \rightarrow (\forall x. P(x)) \rightarrow (\forall x. Q(x))$.
- Give the λP -term corresponding to the formula in 2a.
- Give a closed inhabitant in λP of the answer to 2b.

Exercise 3. (5+3+5 points)

This exercise is concerned with second-order propositional logic (`prop2`) and polymorphic λ -calculus ($\lambda 2$).

- Show that the following formula is a tautology of minimal `prop2`:
 $(\forall c. ((a \rightarrow b \rightarrow c) \rightarrow c)) \rightarrow a.$
- Give the $\lambda 2$ -term corresponding to the formula in 3a.
- Give a closed inhabitant in $\lambda 2$ of the answer to 3b.

Exercise 4. (5+5+5 points)

This exercise is concerned with encodings.

- Consider the following encoding of falsum in `prop2`: `false := $\forall a. a.$`
Show that $\forall c. \text{false} \rightarrow c$ is a tautology of `prop2`.
- Consider the following encoding of disjunction in $\lambda 2$:

$$\text{or } A B := \Pi c : *. (A \rightarrow c) \rightarrow (B \rightarrow c) \rightarrow c$$

Use a term $M : A$ to construct an inhabitant in $\lambda 2$ of `or A B`.

- Consider the following encoding of the datatype of natural numbers in $\lambda 2$:

$$\text{Nat} := \Pi a : *. a \rightarrow (a \rightarrow a) \rightarrow a$$

Give two different inhabitants in $\lambda 2$ of this type.

Exercise 5. (3+4+5+5 points)

Consider the inductive definition the data-type of lists of natural numbers in `Coq`:

```
Inductive natlist : Set :=
| nil : natlist
| cons : nat -> natlist -> natlist.
```

- Give the definition of a function `append : natlist \rightarrow natlist \rightarrow natlist` that appends two `natlists`.
- Give the definition of a function `reverse : natlist \rightarrow natlist` that reverses a list.
- Give the induction principle for `natlist`.
- Give the definition of an inductive data-type `natpair` for pairs of `nats`.

Exercise 6. (5+5+3 points)

This exercise is concerned with inductive predicates in Coq.

- a. Consider the following inductive predicates:

```
Inductive ev : nat -> Prop :=
| ev0 : ev 0
| evS : forall n:nat , odd n -> ev (S n)
with odd : nat -> Prop :=
| oddS : forall n:nat , ev n -> odd (S n) .
```

Give inhabitants of the following:

```
ev 0
odd 1
ev 2
```

- b. Give the definition of an inductive predicate `Inductive threefold: nat -> Prop` which is provable exactly if the input can be divided by 3.
- c. Give an inhabitant of `threefold 0` and of `threefold (S (S (S 0)))`.

The note for the exam is (the total amount of points plus 10) divided by 10.